Speech Understanding
Programming Assignment 3
Report

[Question 1] Audio Deepfake Detection
Goal: The task is to classify the audio samples into Real and Fake.

(Task 1) Use the SSL_W2V model trained for LA and DF tracks of the ASVSpoof dataset. You can find the pre-trained models on this link. Download the custom dataset from here. Report the AUC and EER on this dataset.

Now, as per the instructions, I have used the SSL_W2V model trained for LA and DF tracks of the ASVSpoof dataset. Also, I have downloaded the custom dataset from the provided link. The respective AUC and EER metric of the SSL_W2V pretrained model on the custom dataset is as follows:

Metric name	Metric values
AUC	0.5149
EER	0.4712

(Task 2) Analyze the performance of the model.
Now, observations and respective conclusions for the above mentioned results are as follows:
The performance of the SSL_W2V model on the custom dataset yields an AUC of 0.5149 and an EER of 0.4712. These metrics indicate that the model struggles to effectively distinguish between real and fake audio samples. An AUC value of 0.5149 is only marginally better than random guessing (0.5), suggesting that the model has limited discriminative power on this dataset. The EER of 0.4712 also points to a lack of precision in balancing false positives and false negatives, implying that the model is not making reliable classifications.
The underlying reason for the model's subpar performance could be the mismatch between the training data and the custom dataset. While the model was pre-trained on the LA and DF tracks of the ASVSpoof dataset, the custom dataset may present different types of audio manipulations or conditions that the model was not exposed to during training. This discrepancy can lead to decreased performance due to the model's inability to generalize well to the new dataset.
To improve the model's performance, fine-tuning with the custom dataset is recommended. This process would allow the model to adapt to the specific characteristics and challenges presented by the custom dataset, enhancing its ability to classify audio samples more accurately. Exploring alternative model architectures or additional training data might also contribute to better performance.

(Task 3) Finetune the model on FOR dataset.

Now, as per instructions, I downloaded the FOR dataset from the given link and finetune the above mentioned model on the same. The respective loss and accuracy metrics are reported as follows:

Set	Loss	Accuracy
Training	0.0365	0.9701
Validation	0.1141	0.9362
Testing	0.6097	0.7215

The evaluation of the SSL_W2V model on the FOR dataset after fine-tuning provides insights into the model's performance during training, validation, and testing phases. During training, the model achieved a loss of 0.0365 and an accuracy of 0.9701, indicating that it effectively learned the data, achieving high accuracy with low loss. This suggests that the model has become proficient in recognizing patterns and distinguishing between real and fake audio samples in the training set.

However, the validation set results reveal a loss of 0.1141 and an accuracy of 0.9362, which is slightly lower than the training set results. This discrepancy suggests the model may be starting to overfit, focusing too much on the training data's specific patterns and potentially failing to generalize well to the new, unseen validation data.

The testing phase shows a substantial increase in loss (0.6097) and a decrease in accuracy (0.7215), indicating the model's performance on the testing data is not as strong as on the training and validation sets. The higher loss and lower accuracy suggest that the model may struggle to generalize well to the test data, perhaps due to differences between the training and testing datasets or overfitting during the training phase.

In conclusion, while the model performs well during training, its performance diminishes during validation and testing, indicating issues with overfitting and generalization. To improve the model's performance on unseen data, strategies such as early stopping, data augmentation, or regularization techniques may be beneficial during training to enhance the model's robustness and generalization capabilities.

(Task 4) Report the performance using AUC and EER on FOR dataset.

Now, as per instructions, the performance using AUC and EER on FOR test set is as follows:

Metric name	Metric values
AUC	0.7301
EER	0.4156

The evaluation of the SSL_W2V model on the FOR test set shows an AUC of 0.7301 and an EER of 0.4156. An AUC of 0.7301 suggests that the model demonstrates moderate ability to distinguish between real and fake audio samples, but it is not achieving optimal performance. This value is well above the chance level of 0.5, but it leaves room for improvement in the model's classification capabilities. An EER of 0.4156 indicates that the model experiences a relatively high rate of classification errors, including both false positives and false negatives. The moderate performance may be attributed to the model's training process and the characteristics of the FOR test set. The model may not be fully capturing the specific nuances and intricacies of the FOR test set during the training phase. Additionally, there could be challenges in the diversity and quality of the data used for fine-tuning, which can impact the model's generalization ability.

In conclusion, while the model shows some level of capability in distinguishing between real and fake audio samples on the FOR test set, the AUC and EER metrics indicate that there is significant room for improvement. Further fine-tuning with a more comprehensive dataset or additional training iterations may help the model achieve better performance on the FOR test set.

(Task 5) Use the model trained on the FOR dataset to evaluate the custom dataset. Report the EER and AUC.

Now, as per instructions, I used the model trained on the FOR dataset to evaluate the custom dataset. The respective EER and AUC values are reported as follows:

Metric name	Metric values
AUC	0.8443
EER	0.2011

After fine-tuning the SSL_W2V model on the FOR dataset and re-evaluating it on the custom dataset, the model showed significant improvement with an AUC of 0.8443 and an EER of 0.2011. The AUC value of 0.8443 indicates a strong ability to distinguish between real and fake audio samples, demonstrating the model's improved discriminative power. An EER of 0.2011 suggests a substantial reduction in both false positives and false negatives, reflecting more precise and balanced classification.

The fine-tuning process on the FOR dataset likely enabled the model to better adapt to the nuances of the custom dataset. The FOR dataset may contain audio samples and manipulations more similar to those in the custom dataset, allowing the model to learn features and patterns that are more relevant and transferrable. This improved generalization is evident in the marked enhancement in the model's performance metrics.

In conclusion, fine-tuning the SSL_W2V model on the FOR dataset proved highly beneficial for the model's ability to classify the custom dataset accurately. The significant increase in AUC and decrease in EER highlight the effectiveness of this approach. To maintain and further improve performance, continued fine-tuning with diverse datasets that align with the target data's characteristics is recommended.

(Task 6) Comment on the change in performance, if any.

Now, the metrics for the pretrained and finetuned SSL_W2V model on the custom dataset are as follows:

Metric name	Pretrained	Fine Tuned
AUC	0.5149	0.8443
EER	0.4712	0.2011

Comparing the performance of the SSL_W2V model on the custom dataset before and after fine-tuning on the FOR dataset demonstrates a significant improvement in the model's classification capabilities. Initially, the pre-trained model achieved an AUC of 0.5149 and an EER of 0.4712, indicative of near-random performance. This lack of effectiveness was likely due to a mismatch between the ASVSpoof dataset used for pre-training and the custom dataset, as the former may not encompass the full range of audio manipulations and characteristics present in the latter.

After fine-tuning the model on the FOR dataset, its performance improved substantially, with an AUC of 0.8443 and an EER of 0.2011 on the custom dataset. Fine-tuning allowed the model to adapt to the specific features and patterns of the FOR dataset, which may be more representative of the custom dataset. This adjustment in training data helped the model learn more nuanced audio characteristics and manipulations, enhancing its ability to generalize and classify audio samples more accurately.

Additionally, the fine-tuning process likely optimized the model's weights and biases, honing its focus on relevant features and reducing noise and irrelevant patterns. As a result, the model became more sensitive to the subtle differences between real and fake audio samples, leading to a notable improvement in both AUC and EER metrics on the custom dataset.

In conclusion, the substantial enhancement in performance after fine-tuning can be attributed to the closer alignment between the training dataset (FOR) and the custom dataset, as well as the refinement of the model's learning process. This emphasizes the importance of training the model on data that closely resembles the target dataset for achieving better classification outcomes.

NOTE: Please find attached the Github link for the code.